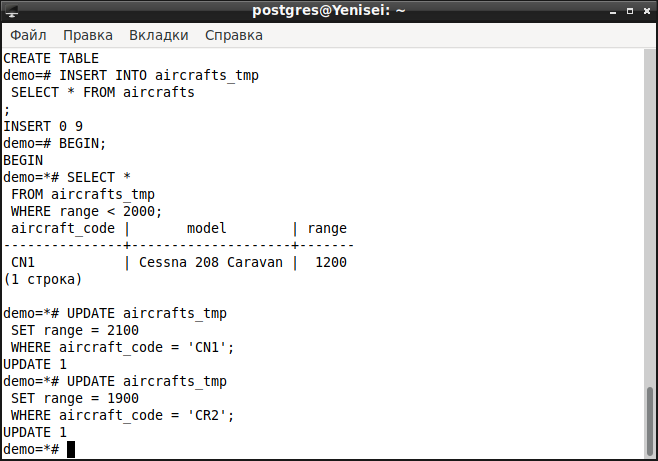
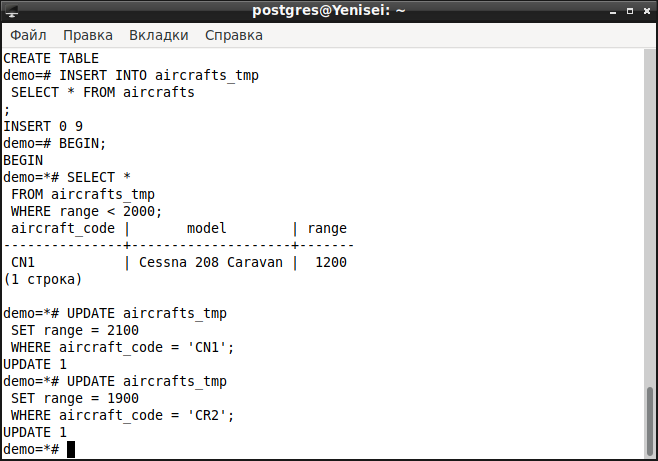


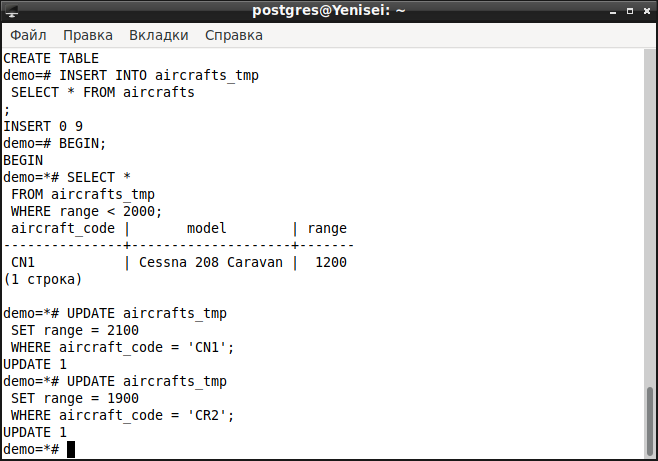
*Начнем транзакцию на первом терминале.*



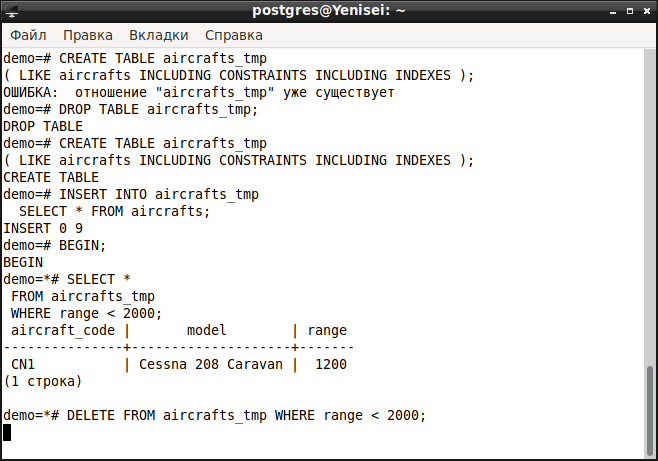
*Выберем все самолеты с дальностью полета меньше 2000.*



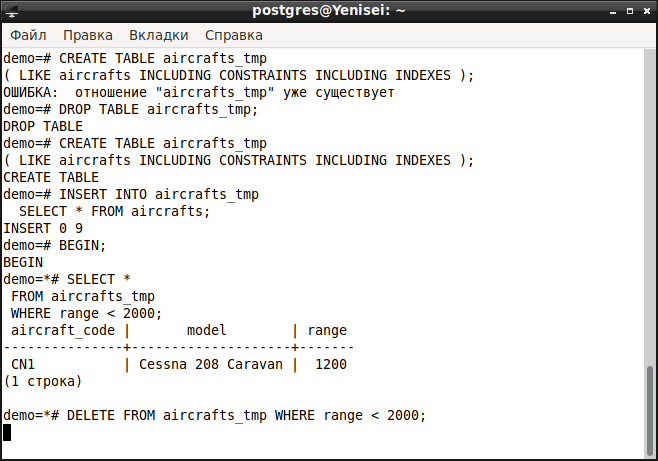
*Обновим данные для модели Cessna 208 Caravan.*



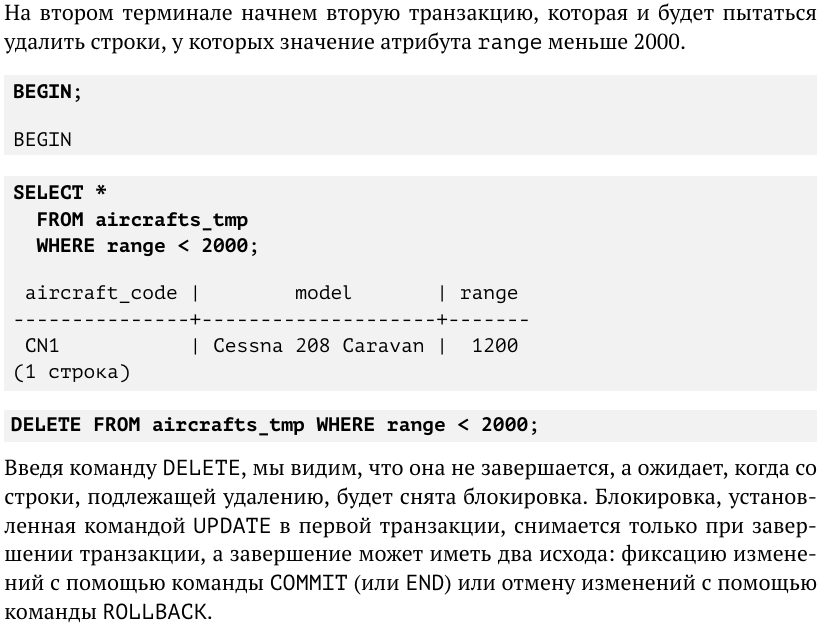
*А также для модели Bombardier CRJ-200.*



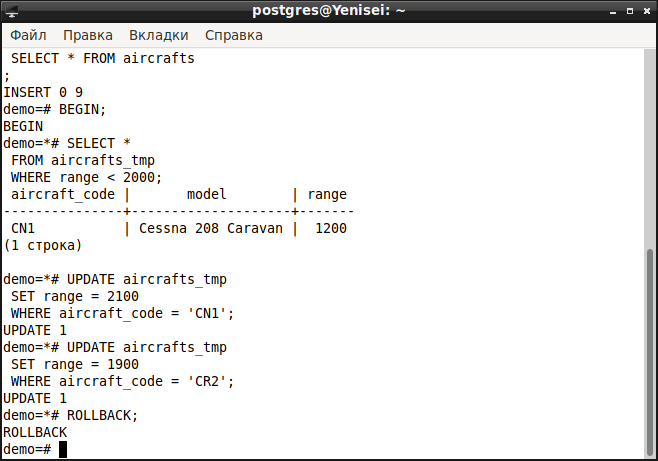
*На втором терминале начнем вторую транзакцию.*



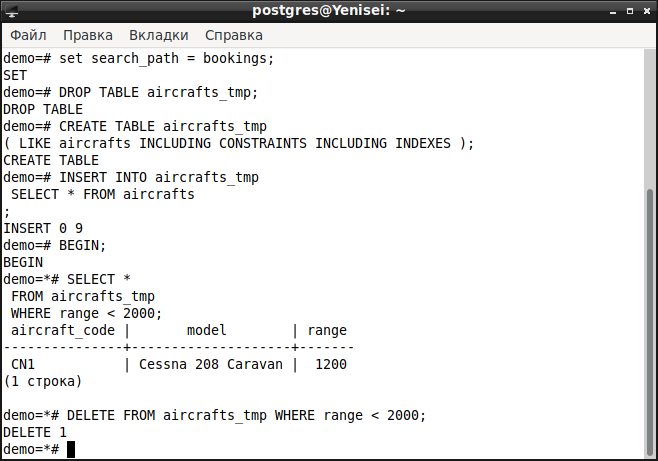
*Также выберем все самолеты с дальностью полета меньше 2000.*



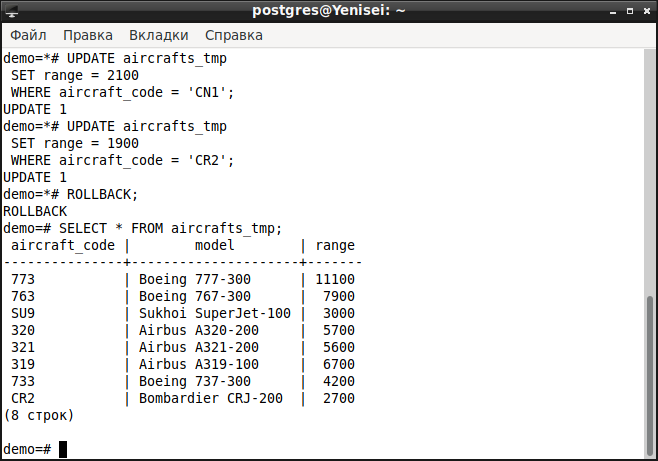
*Удалим самолеты с дальностью полета < 2000;*

**

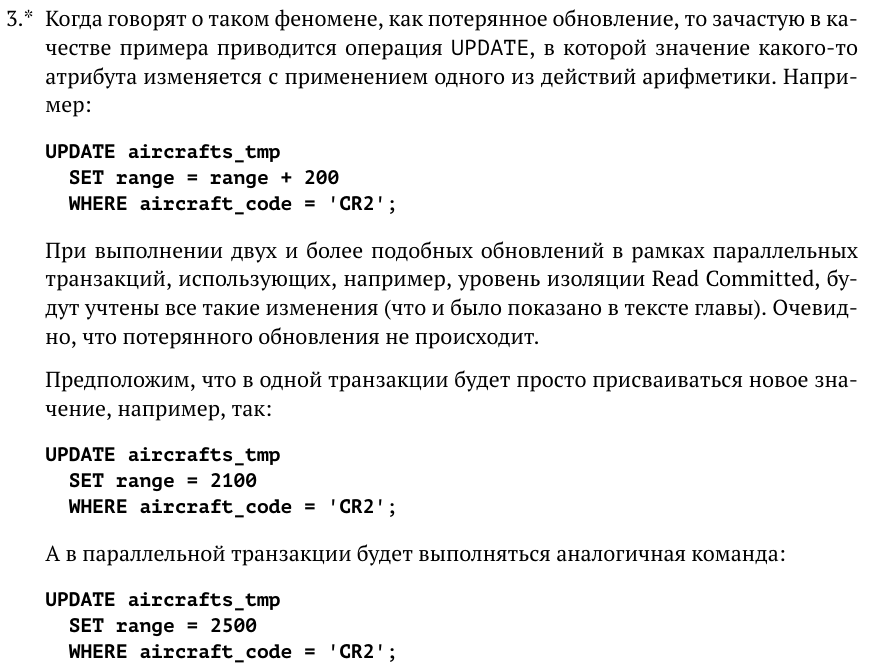
*На первом терминале откатим транзакцию.*

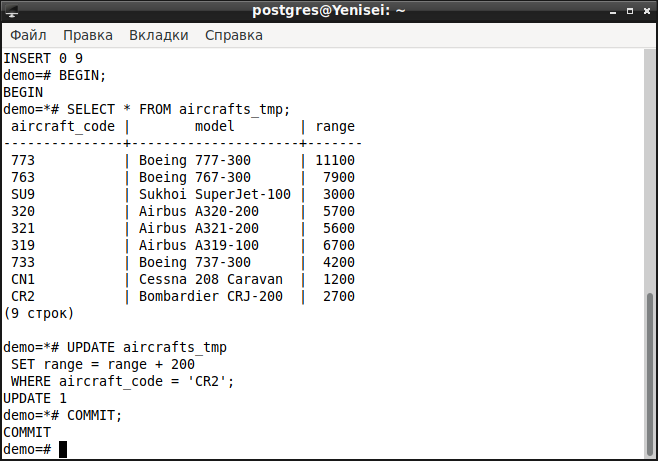
**

*На втором терминале было успешно произведено удаление некоторой строки.*

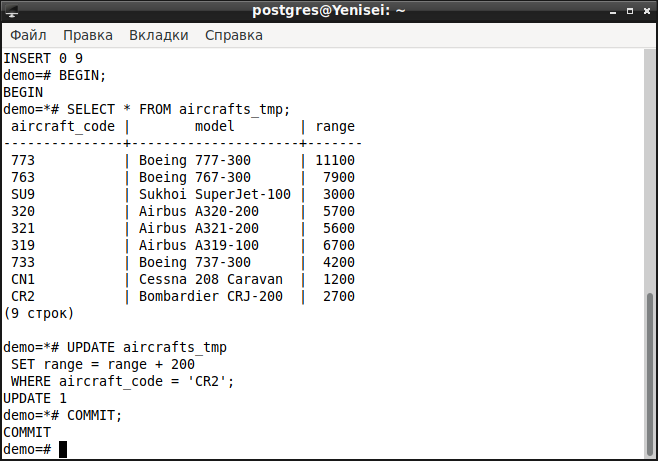
**

*Просмотрев содержимое таблицы aircrafts\_tmp убедимся, что была удалена модель Cessna 208 Caravan. Это происходит, потому что при откате изменений в первой транзакции таблица вернулась к первоначальному состоянию и строка содержащая модель Cessna 208 Caravan вновь стала подходить под критерии удаления. Команда DELETE вновь прочитала содержащую эту модель строку и убедившись в том, что строка подходит под критерии, удалила ее.*

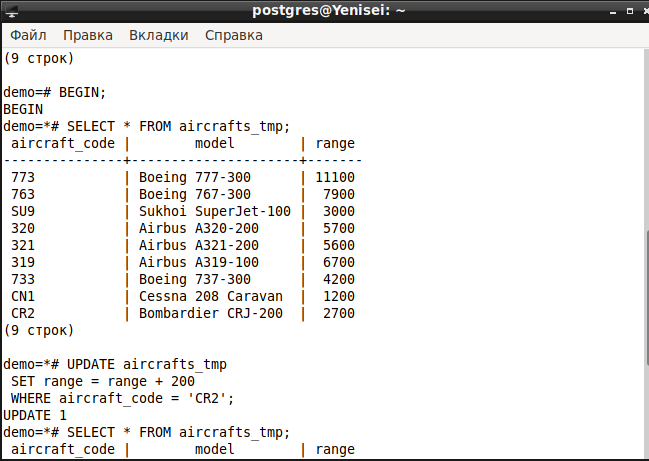
**

**

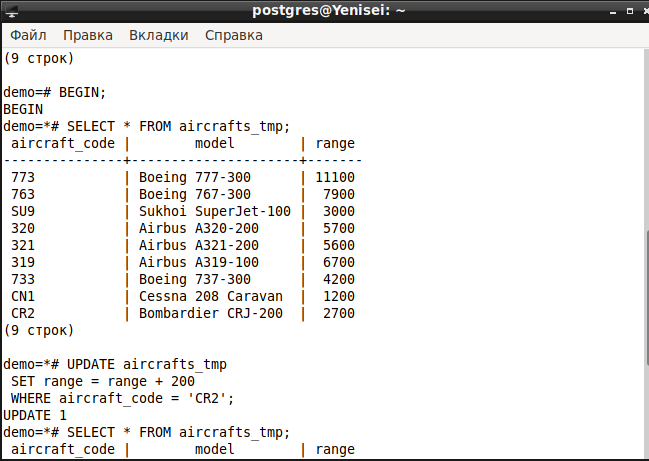
*Начнем операцию на первом терминале и проверим состояние таблицы.*

**

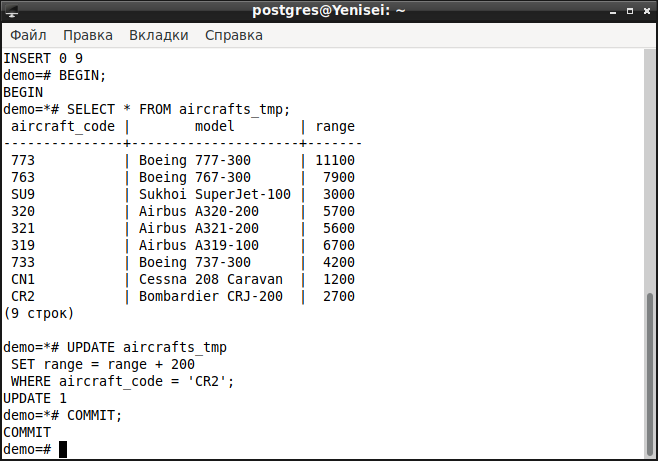
*Обновим данные для aircraft\_code = ‘CR2’.*

**

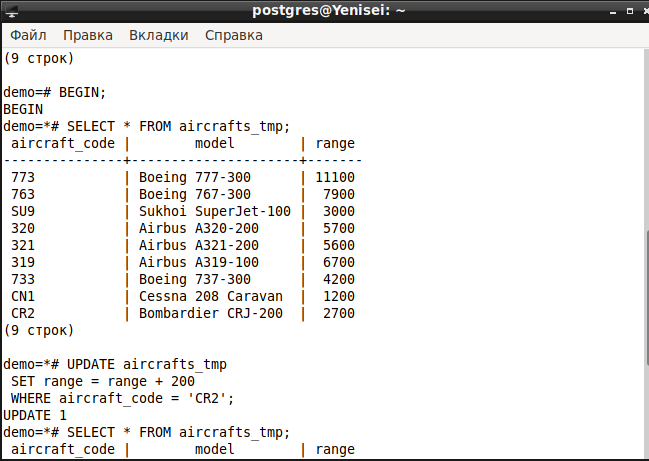
*Начнем транзакцию на втором терминале.*

**

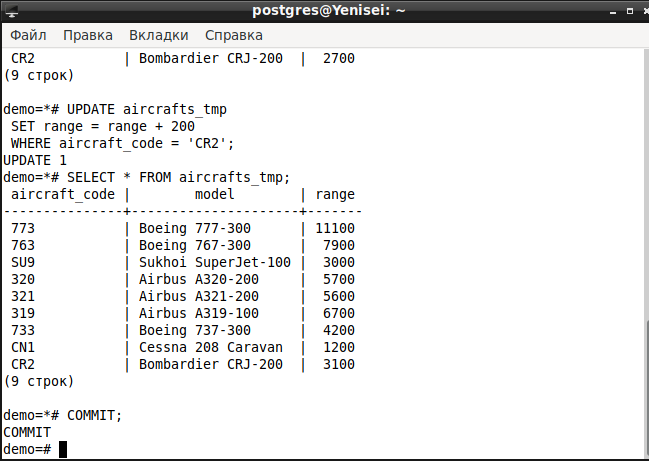
*Обновим данные в той же строке. Обновление повиснет в ожидании окончания транзакции в первом терминале.*

**

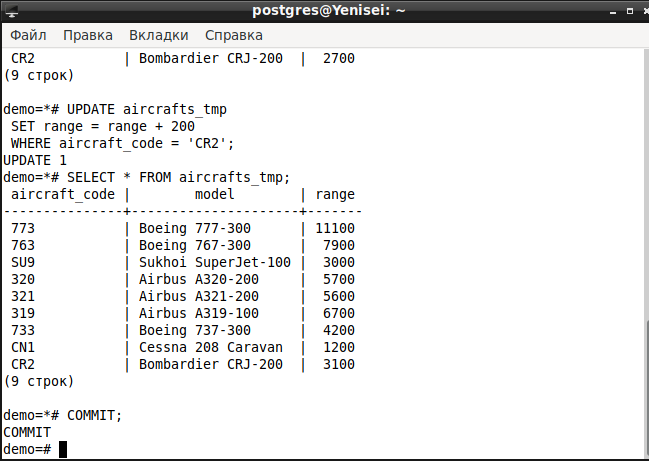
*Завершим транзакцию в первом терминале.*

**

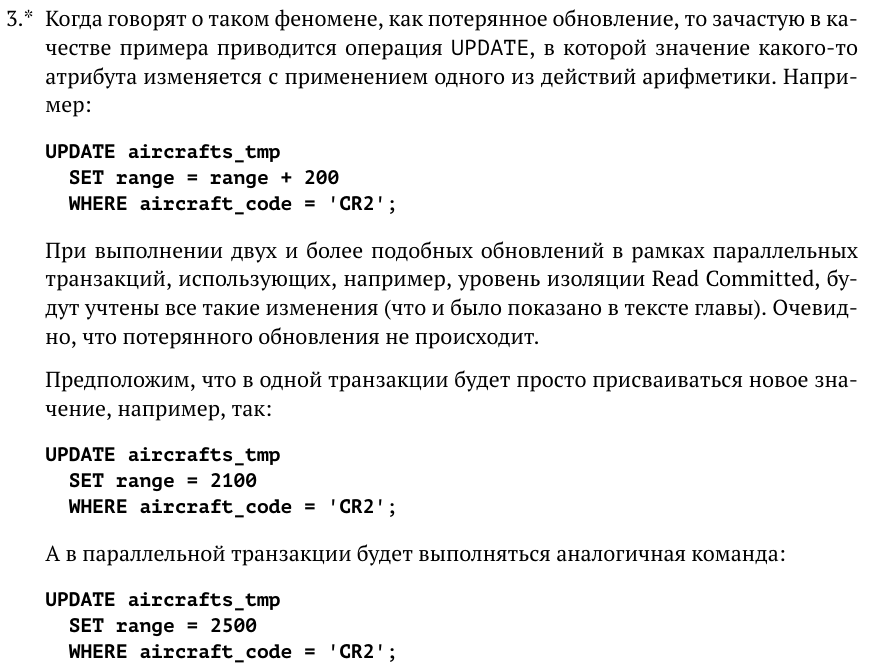
*Обновление во втором терминале также завершится.*

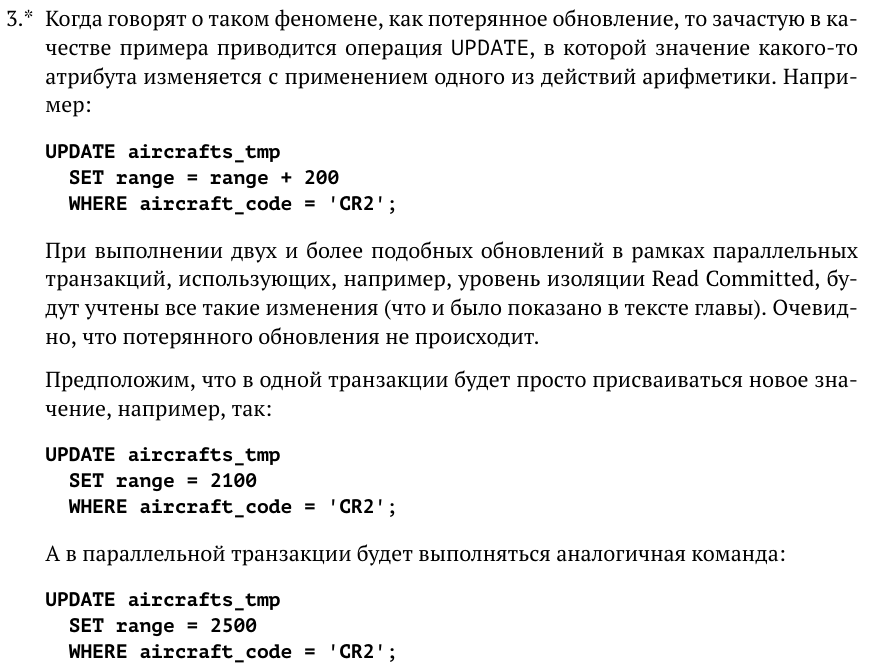
**

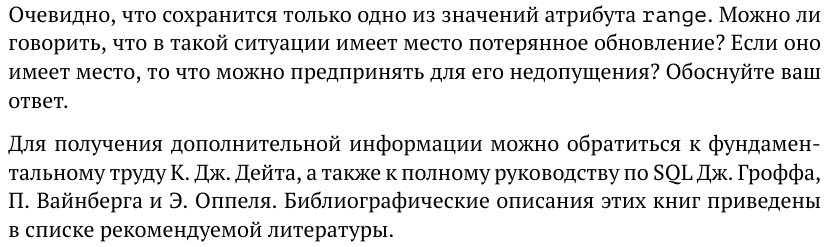
*Выведем таблицу и убедимся, что были произведены оба обновления строки с aircraft\_code = 'CR2'.*

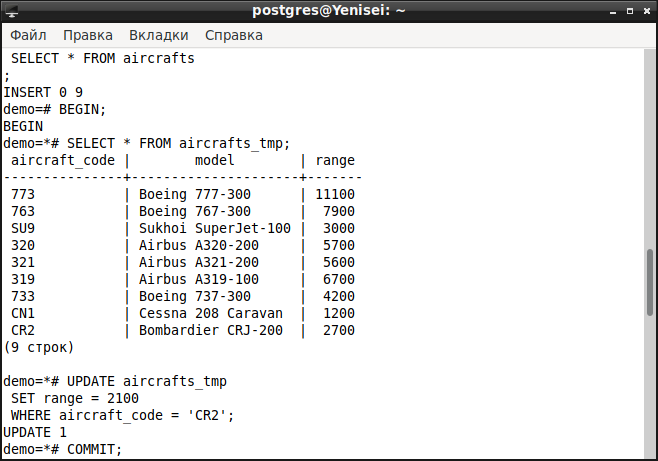
**

*Завершим транзакцию во втором терминале.*

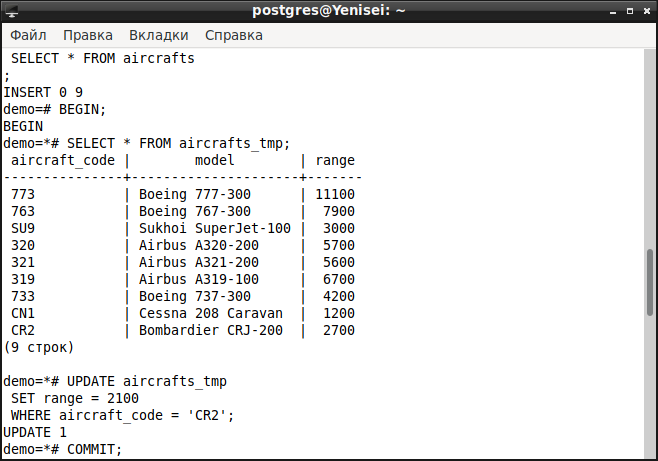
**

**

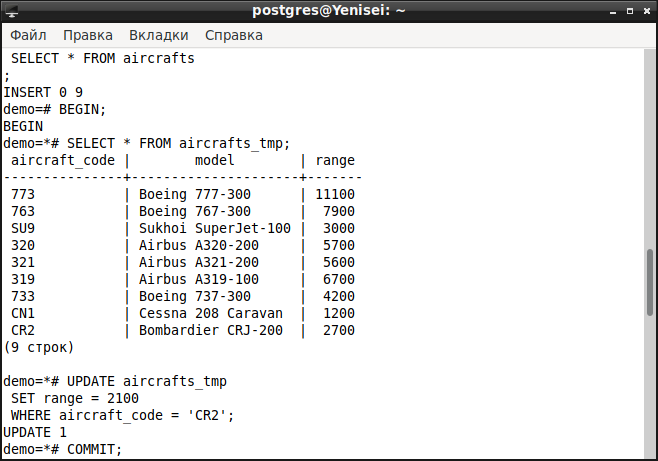
**

**

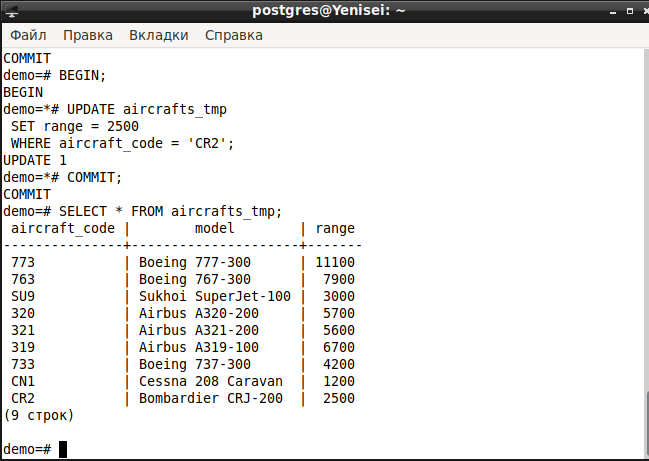
*Запустим транзакцию в первом терминале и посмотрим содержимое таблицы aircrafts\_tmp.*

**

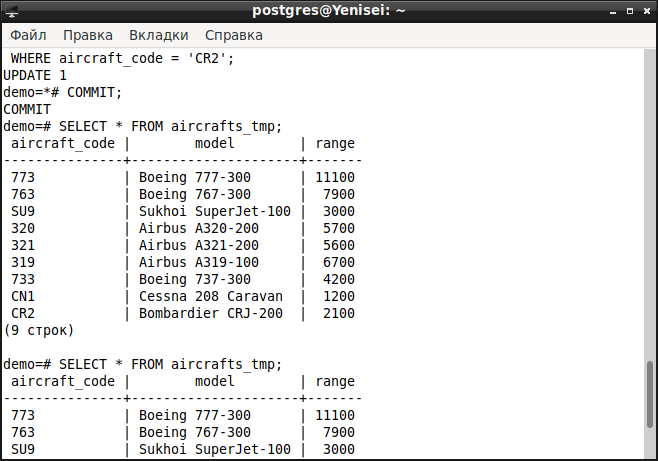
*Обновим данные в таблице aircrafts\_tmp.*

**

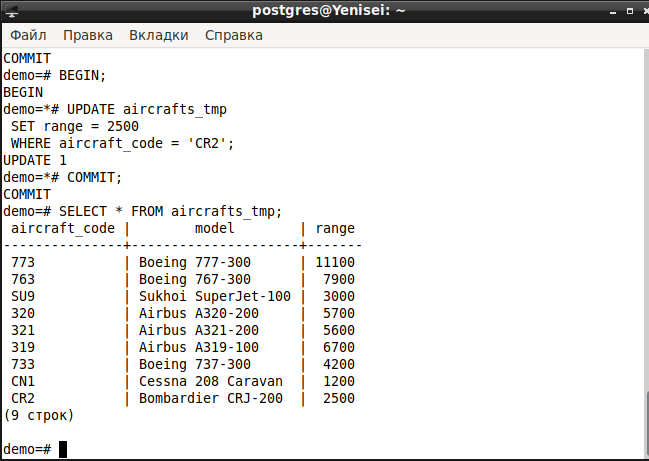
*Во втором терминале также запустим транзакцию и посмотрим на содержимое таблицы aircrafts\_tmp. Как мы видим, содержимое таблицы совпадает с изначальным, так как транзакция на первом терминале пока не была закончена.*

**

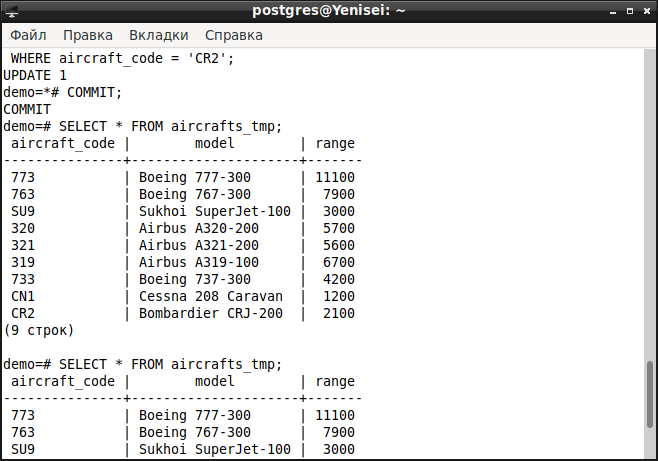
*Обновим данные во втором терминале. Обновление повиснет, ожидая окончания транзакции в первом терминале.*

**

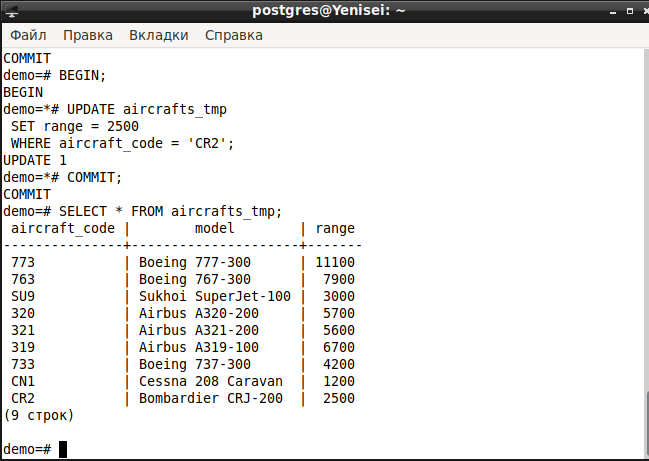
*Завершим первую транзакцию.*

**

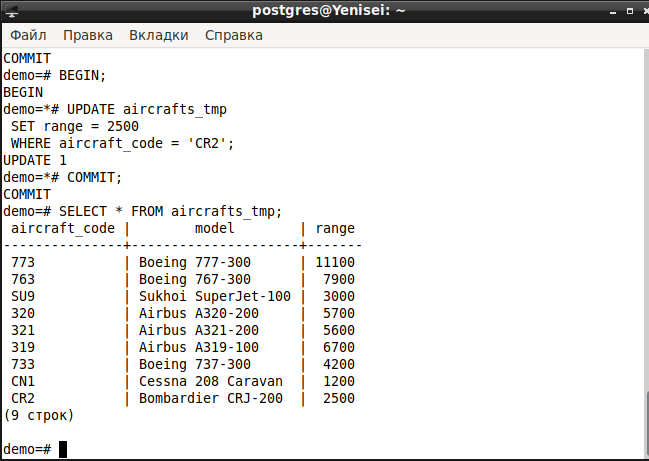
*Строка, содержащая aircraft\_code = ‘CR2’ разблокируется и во втором терминале будет произведено обновление.*

**

*Посмотрим на таблицу в первом терминале. Как мы видим, данные для строки содержащей aircraft\_code = ‘CR2’, обновились согласно действиям, совершенным в первом терминале, но к строке не применено обновление, совершенное во втором терминале. Это происходит потому, что во втором терминале транзакция еще не завершена.*

**

*Завершим транзакцию во втором терминале.*

**

*Теперь для любого терминала, результат выборки всех данных из таблицы покажет обновление строки, содержащей aircraft\_code = ‘CR2’, согласно обновлению, произведенному во втором терминале.*

*Таким образом, нет никаких потерянных обновлений. Есть только последовательное обновление строки.*